Ugo Anyaegbunam, Anne Lehr, Jacky Guzman Nunez

Part 1: Installing a PHP web shell

Answer these questions:

1. Explain how you can execute the Linux command `whoami` on the server using your webshell. What result do you get when you execute that command?
    a. http://danger.jeffondich.com/uploadedimages/anyaegbunamu-webshell.php?command=%22whoami%22 in the url is how I executed it and it returned www-data
2. What is this webshell's <pre> tag for? (And more to the point, what happens if you leave it out?)
    a. It is for formatting text. When I took it out and ran ls all the filenames appeared side by side rather than top down

Part 2: Looking around

Now that you have a webshell working, what can you do with it? Let's find out. Warning: do not mess with your classmates' files!

Answer these questions:

1. What directory is danger's website located in?
    a. A directory called www
2. What are the names of all the user accounts on danger.jeffondich.com? How do you know?
    a. Jeff and bullwinkle because those were the only 2 directories in the home directory and thats the parent directory of user accounts on devices.
3. Do you have access to the file /etc/passwd? What's in it?

a. Yes, it has information on all the user accounts:

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:102:105::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:103:106:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
syslog:x:104:111::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:112:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:113::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:114::/nonexistent:/usr/sbin/nologin
usbmux:x:109:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
sshd:x:110:65534::/run/sshd:/usr/sbin/nologin
pollinate:x:111:1::/var/cache/pollinate:/bin/false
landscape:x:112:116::/var/lib/landscape:/usr/sbin/nologin
fwupd-refresh:x:113:117:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin
jeff:x:1000:1000:Jeff Ondich,,,:/home/jeff:/bin/bash
postgres:x:114:119:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
bullwinkle:x:1001:1001:Bullwinkle J. Moose,,,:/home/bullwinkle:/bin/bash
```

4. Do you have access to the file /etc/shadow? What's in it? (You'll have to look onliine for the answer to that second question, since the answer to the first is no.)
    a. No because it’s only available to the root user. It contains encrypted user passwords and other user data
5. There may be some secret files scattered around. See how many you can find and report on your discoveries.

a. Found this:

```
Congratulations!

          _   _
         (.)_(.)
      _ (   _   ) _
     / \/`-----'\/ \
   __\ ( (     ) ) /__
   )   /\ \._./ /\   (
    )_/ /|\   /|\ \_(

by Joan Stark, https://www.asciiart.eu/animals/frogs
```

b. And this:

```
Congratulations!

  ___     ___
 (o o)   (o o)
(  V  ) (  V  )
/--m-m- /--m-m-

https://www.asciiart.eu/animals/birds-land
```

c. Found this looking in Jeff directory:

```
supersecret.txt below
Congratulations!

/   \          /   \
\_   \        /  __/
 _\   \      /  /__
 \___  \____/   __/
     \_       _/
       | @ @  \_
       |
      _/     /\
     /o)  (o/\ \_
     \_____/ /
       \____/

https://www.asciiart.eu/animals/moose

youfoundme.txt below
This isn't the secret, but hi!
```

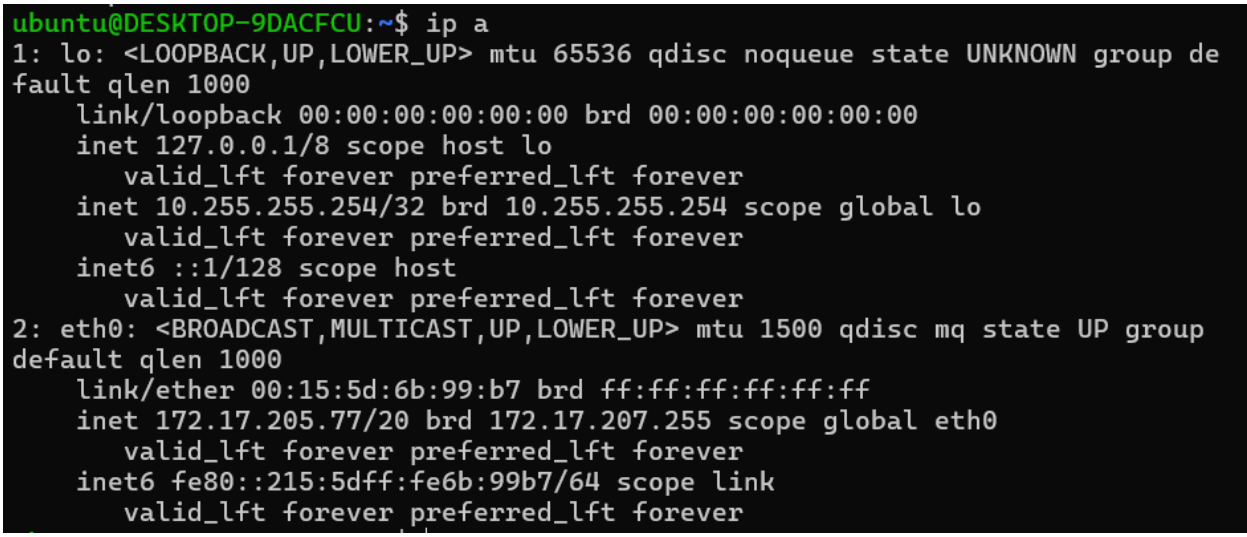
6. [Optional] Report on anything else interesting you discover.

a. The lost+found folder in the home directory has everything ever uploaded, including the webshells
b. Bullwinkle has everyone's uploads as well
c. Jeff has the attacks and uploads form 2023 in his directory. I'm sure I could curl -O them if I was that curious.

Part 4: launching a reverse shell

Answer these questions and do these things:

1. What is the IP address of your Kali VM (the target machine)? How did you find out?
    a. 192.168.16.128 and I know because that is the ip we've been using in the curl request. I also ran curl -v 'http://192.168.16.128/anyaegbunamu-webshell.php?command=ifconfig' to see the IP as well
2. What are the IP addresses of your host OS (the attacking machine)? How did you find out? Which one should you use to communicate with Kali and why?
    a. I ran ip a on my host machine and got:



    I believe we use the first inet ip under eth0 and we always have for all our labs. It's also the only one that looked remotely correct to me. On top of that, according to the results of my google search it says "The correct Class A IP address range is 0–127, with the host address formatted as xxx.xxx.xxx. "
3. On your host OS (the attacker), pick any port number between 5000 and 10000

4. Go back and look at your `nc -l -p` terminal on your host OS (attacking machine). Do you have a shell now? Is it letting you execute commands on Kali? How do you know it's Kali?
    a. This is wild. I do have a shell now and it is letting me execute commands. I know it is kali because I'm in the directory for the web server being run on the kali machine
5. What are all those % codes in the URL you used?
    a. They're a way to represent characters that aren't typically parsed in URLs. For example, all the spaces in my commands ran with the webshell are converted to %20
6. Write a brief description, probably including a diagram, explaining how this reverse shell is functioning.
    a. The attacker sets up a listening session on their machine tuned to the IP of the target server/machine. From there, a web shell is placed/forced onto the server via some file transfer or whatever. We put it there ourselves, but I don't know how I would do it in real life. From there, we send this big boy bash command via the command parameter in the curl request to the target machine. From what little research I've done and what I viewed during the assignment, it looks like it tells the target computer to start a new interactive shell and redirect the stdout and stdin to correlate with the newly initiated TCP connection that came from our initial curl. So then stdin becomes what we send in the listening session, and the output is received on our end by the listening session. The whole time, though, we're tapped in to the bash session happening on their machine.